

REMARKS

Status of Claims:

Claims 1-31 are pending.

Prior Art Rejection:

Claims 1-31 stand rejected under 35 USC 103 as unpatentable over Putzolu et al in view of Takewaki et al.

The examiner's rejections are respectfully traversed.

Argument:

A. General

It is the applicant's position to explain why the cited references are remote from the present invention.

(1) Summary of Putzolu

Putzolu relates to network environment supporting mobile agents with permissions access to resources, in other words, a method and system for providing an environment allowing agents to function on a set of devices having resources, the environment providing services allowing agents to access to resources. (Abstract)

Particularly, there is disclosed a system for controlling resources of a computer or the like (PC or routers) at the moved destination of the mobile agent. The mobile agent migrates from an external computer to a computer in which the resource is present, and thus the access to the resource is restricted by using a security model such as a sandbox etc. According to Putzolu, a service object is provided for getting access to resources, so that a mobile agent can indirectly access the resources. Also, in order to secure the security for the access restriction to the resources, the agent has an access control list, based on which permission

for movement, use of resource object and permission of use of a method of the resource object is determined.

(2) Summary of Embodiments of the invention

The present invention relates to a shell agent having interpreter of the script language provided on the mobile agent, while it is based on the mobile agent technology. The mobile agent can get access to resources at the movement-destination, whereat, however, the directory is unspecified and it is necessary for the file to identify the file by an absolute path within a computer at the movement-destination.

According to the present invention, it is proposed how to handle the file resources with ease, i. e., "shell agent". The shell agent has an interpreter of script language and can change a work directory by cd (change directory). As for the usual shell, the directory is generally closed within a computer. However, according to the present invention, it is possible to perform "cd" to a directory belonging to another different computer by utilizing the global directory.

If "cd" is performed on a directory of another computer, the shell agent itself moves, as a mobile agent, to the computer that has the directory.

It should be noted that the shell agent itself does not move in the interactive mode, and the remote resources are utilized, using the remote shell calling function, without movement of the shell agent per se.

Also, the shell agent according to the present invention has a parallel execution function, in which parallel applications can be executed in parallel in the plurality of computers.

(3) Difference Between Embodiments of the Invention and Putzola

Putzolu: a program, in which a mobile agent per se, moves and gets access to a remote resources.

Present invention: a Script language is provided on a mobile agent, a shell agent for interpreting the script language is made, with the shell agent performing `cd` during interpreting the script, thereby performing movement to a remote computer, resulting in access to the remote resource.

In short, the difference resides in the mobile agents per se for Putzolu, and the provision of the script language on the mobile agent for the present invention.

It is necessary to make a programming beforehand in order to accomplish movement and access to resources by using the mobile agent as it be. By using the script language, the description is simple and concise, and may have additional advantage that allows interactively input access command to the resources.

Functionally, there is a difference in Putzolu, in which the mobile agent gets access to a file using a local directory local for a computer concerned at the destination of movement. In contrast, the present invention allows an access to a file identified by a global directory that is under control over a plurality of computers in the script language on the mobile agent.

The mobile agent of Putzolu has no concept of the work directory immediately after the movement. According to the present invention, the work directory is identified at any time using the global directory. And thus, a certain directory identified on a remote computer is also controlled also after movement as a work directory.

By explicitly controlling the directory of the shell agent, operation onto the remote directory and file resources are simplified.

The worksheet system is disclosed by Putzolu for controlling an execution list at the destination of movement. However, this system only controls the list for performing event processing at the destination of movement. This is totally different from the present invention. Furthermore, using this worksheet, the processing accompanied by the work directory is impossible, which is also counted as a difference from the present invention.

Further, parallel execution is never mentioned by Putzolu. In contrast, the present invention enables it based on the claimed feature.

B. Examiner's Reliance on Putzolu

The rejection primarily relies on Putzolu stating that Putzolu discloses remote resources using moving (mobile) agent(s), and is obvious in view of Takewaki.

However, the examiner's reading/interpretation of Putzolu is incorrect. For instance, the examiner asserts that Putzolu discloses the feature (a1) (§2, page 2)

(a1) a shell agent responsive to an input of a script language
... (col. 11, lines 4-26, line 67, col. 12, lines 1-14 and lines 49-66);

However such "shell agent responsive to an input of a script language" does not exist in the system of Putzolu. The passages cited by the examiner are not related to (a1) of the present invention, and the examiner is simply in error.

Putzolu relates to the invention directed to mobile agents per se, whereas the present invention is directed to the script language implemented on the mobile agent.

C. Re: Putzolu as Applied Against Claims 1, 11, 21 and 22

-- (a) shell agent responsive to an input of a script language controlling the distributed application to interpret and execute said script language --

(1) Putzolu discloses merely a management console application, which is not related with the shell agent.

Putzolu, at col. 11, lines 4-26, discloses a "worksheet", which, however, is a mere object table for storing a plurality of execution codes such that a code of a suitable worksheet object is executed responsive to an event (input). This is only the execution mechanism of the mobile agent, and has nothing to do with the shell agent responsive to input of script language.

The artisan can not interpret this worksheet as the shell agent responsive to input (from

outside) of script language?

The examiner might misunderstand the "Java language worksheet" as being a "script language", which, however, "is an object containing code executed by a service." (col. 11, line 13-15)

This disclosure relates to an execution mechanism of the mobile agent, and has nothing to do with the script language implemented on the mobile agent, nor an INTERPRETER.

Further, the term "worksheet" has no concept such as the case with the script language having a CURRENT DIRECTORY such that commands are executed in sequence.

- (2) In the present invention, being provided with the script language under the directory-control, it is essential to control mobile agent(s) and the file directory.

Refer to Figs. 3 and 4, the shell agent has SHELL INTERPRETER 20 having the shell interpreter to interpret and execute the script language and invokes the agent generator (16). (Refer to [0033], [0037], [0038] etc.)

This feature is simply not found in Putzolu.

Thus difference of claim 1, (a1) and claim 11(a) is readily distinguishes the present invention from Putzolu.

- (3) Further Difference Over Putzolu

- (3.1) Putzolu does not disclose the concept of "global directory", instead only a "file object".

The presence of the global directory is important in certain embodiments of the invention, to give a model wherein moving is executed when its work directory is present in a remote computer.

- (3.2) "Current directory 24, Fig. 3",

The presence or non-presence of this feature is also important. This is defined by claims 2 and 12.

- (4) Claims 2 and 12: current directory

Putzolu, col. 15, lines 36-54:

This passage discloses merely that a file access can be performed by an object having a common interface. This merely explains the "File Object" of Java. Claims 2 and 12 explicitly define the presence of the current directory, which enables relative file access

from the current directory. Also the file directory of embodiments of the present invention is expressed by a global directory which spans (or covers) a plurality of computers.

Putzolu, on the other hand, merely mentions that local directory name and file name at the destination of movement can be used in identifying the file.

As to the shell interpreter, col. 11, lines 4-26 of Putzolu has nothing to do with this as there is no interpreter of the script.

(5) Claims 3 and 13

Putzolu merely discloses that the access interfaces are common per every resource (device).

In embodiments of the present invention a repository path table is defined.

The repository is provided for storing and managing the manner for retrieving (calling) command from the script language. In the repository, a variety of applications are registered and not limited to the resource controlling.

(6) Claims 4 and 14: parallel execution

Putzolu does not disclose parallel execution. Particularly, there is no disclosure on how the parallel execution is performed for controlling a plurality of computers.

(7) Claims 5 and 15:

The local service agent is employed in order to make the relation between movable shell agent and non-movable application agent an indirect reference-relationship.

Putzolu discloses that a mobile agent calls a service. According to Putzolu, the resource control command is not executed by using three agents: shell agent, local service agent and resource control command.

If any correspondence be assumed with embodiments of the present invention, it would be as follows.

present invention	Putzolu
shell agent	mobile agent
application agent	service

(8) Claims 6 and 16: Agent referencing table

This is a controlling table for controlling the instance of the invoked application agent.

Thus, this is different in nature from the service controlling manner which controls per device, or the code table, at the pre-execution state, registered in the worksheet.

(9) Claims 7 and 17

Main claims 1 and 11 are different.

(10) Claims 8 and 18

These claims are defined in order to cover the function of identifying an application to be invoked by extension of the file, i.e., due to the script language.

Putzolu does not disclose anything on the file extension.

(11) Claims 9 and 19

Putzolu does not disclose the parallel execution, even touching on thread, at all.

(12) Claims 10 and 20: Subshell agent

Subshell agent is utilized in the present invention, in order to achieve the parallel execution over a plurality of computers.

Putzolu does not mention anything on this, and thus there is no subshell agent.

(13) Claim 23: as claims 1 and 11

(14) Claim 24:

Putzolu does not have script language nor perform the parallel execution (processing).

(15) Claim 25

Putzolu discloses no concept of current directory. Since Putzolu has no script language, there is no explanation on the interpreter for command execution.

(16) Claim 26

Putzolu does not disclose how to control (or manage) the service (i.e., corresponds to the "application agent" of the present invention).

According to the present invention, claim 26 sets forth how the location where the application exists is identified through the repository path (i.e., path information stored in the repository path table).

(17) Claim 27: cd (change directory)

Putzolu does not disclose any concept that the directory movement and RPC are determined by cd.

In claim 27, it is determined whether the destination of cd is local directory or remote directory. Also it is determined whether parallel execution takes place depending on

the parallel execution counter (i.e., terminal mode flag). During parallel execution of the shell agent, the shell agent is not moved between computers; and also it is determined whether if moved, to be moved in a batch mode or not or moved in an interactive mode, etc.

Such disclosure is not found in Putzolu.

(18) Claim 28: parallel execution

This feature is not disclosed by Putzolu.

(19) Claim 29: current directory

Based on the presence of the current directory, it is necessary to compare whether an execution command is present in a directory that is the same as the current directory. Putzolu has no such concept of the current directory, and thus there is no need of the function defined by claim 29.

(20) Claim 30

As claim 29

(21) Claim 31

Based on the script language, this claim is established in order to identify an application-to-be-invoked by means of file extension.

On the other hand Putzolu does not disclose the file extension, at all.

Conclusion:

Applicant believes that the present application is now in condition for allowance. Favorable reconsideration of the application as amended is respectfully requested.

The Examiner is invited to contact the undersigned by telephone if it is felt that a telephone interview would advance the prosecution of the present application.

The Commissioner is hereby authorized to charge any additional fees which may be required regarding this application under 37 C.F.R. §§ 1.16-1.17, or credit any overpayment, to Deposit Account No. 19-0741. Should no proper payment be enclosed herewith, as by a check being in the wrong amount, unsigned, post-dated, otherwise improper or informal or even entirely missing, the Commissioner is authorized to charge the unpaid amount to

Deposit Account No. 19-0741. If any extensions of time are needed for timely acceptance of papers submitted herewith, Applicant hereby petitions for such extension under 37 C.F.R. §1.136 and authorizes payment of any such extensions fees to Deposit Account No. 19-0741.

Respectfully submitted,

Date 7-30-04

By David A. Blumenthal

FOLEY & LARDNER LLP
Customer Number: 22428
Telephone: (202) 672-5407
Facsimile: (202) 672-5399

David A. Blumenthal
Attorney for Applicant
Registration No. 26,257